1970

Computers Versus Mathematics: Round 2

Cowie, James B.; Fremgen, James M.

Cowie, James B., and James M. Fremgen. "Computers versus Mathematics: Round 2." The Accounting Review 45.1 (1970): 27-37.
http://hdl.handle.net/10945/57534

Computers Versus Mathematics: Round 2

James B. Cowie and James M. Fremgen

In THE ACCOUNTING REVIEW of April 1969, A. Wayne Corcoran matched the computer against mathematics in a fight for the accountant's attention.[1] At the very outset, he acknowledged his bias in favor of mathematics (p. 360). This was neither surprising nor disturbing, as people who are wholly neutral on a subject seldom take the time to write about it. Corcoran, however, went a step beyond bias. He tied the computer's right hand behind its back. As a result, mathematics was declared the winner of what we believe was a "fixed" fight. Our purpose here is to untie the computer's right hand so that it may do itself justice, and this does not involve taking swings at mathematics. The computer and mathematics are not inherently competitors. They contest only for the limited time and talents of the student and the practitioner of accounting (or, for that matter, of any other discipline). Nor are they the only contestants for the accountant's attention. Thus, we are concerned here with only one segment of a narrowly defined problem.

Regrettably, the problem is a real one. No one can be even moderately conversant with—let alone expert in—all areas of knowledge that are in some way relevant to his chosen area of major interest. Even in a 5-year program, a student cannot meet all of the general educational requirements for a degree along with the special requirements for a major in accounting and simultaneously become reasonably proficient in both mathematics and the computer. He will have to settle for fewer courses in these areas than would have potential value to him in his professional career. This unhappy necessity should be mitigated by a systematic program of continuing education. Even this does not change the fact that the accountant will never be as well educated as he would like to be. It is reasonable, then, for him to allocate his scarce personal resources on the basis of a judgment as to the utility of alternative courses of study.

In making his choices, he may have to be somewhat ruthless. Very sound academic material may have to be rejected (or, at least, deferred) in favor of what he has reason to believe will be more useful to him as an accountant. Thus, there are three basic questions that need to be answered before a choice can be made between extra study of computers or mathematics.

1. Why should an accountant study mathematics and what specific

[1] "Computers Versus Mathematics," THE ACCOUNTING REVIEW (April 1969), pp. 359–74. All parenthetical page references in the text of this paper are to Corcoran's article.

James B. Cowie and James M. Fremgen are Associate Professor of Management Science and Professor of Accounting, respectively, at the U. S. Naval Postgraduate School.

mathematical subjects are recommended?
2. Why should an accountant study computers and what should he learn about them?
3. In view of all the other demands on his time and attention, what is a reasonable allocation of effort to each of these areas?

Unfortunately, the third and most critical of these questions is also the most difficult to answer. It requires a balancing of the answers to the first two questions and proper attention to all other contestants for his time. We should not assume that a definitive answer exists, and we would expect different accountants to make different determinations consistent with the requirements of their work. One CPA, for example, may spend much of his time working with operations research models in the management services function and may thus conclude that his most urgent need is to learn more about their underlying mathematics. Another CPA might spend most of his time auditing computer-based accounting systems and might decide to devote what time he can to the study of computers. There is a great deal he might learn about internal controls in a computer system and about the auditing procedures that the computer can perform or facilitate.

The student of accounting does not have his day-to-day responsibilities to guide him in choosing between alternative courses of study. He must rely largely on advice from others, chiefly his instructors. It would be a serious error for any student of accounting today to ignore either the computer or mathematics, and it would be wrong for any instructor to so advise him. The student has a right to expect that the content of his curriculum will be based upon careful thought as to his future needs, not upon faculty whims or what is currently fashionable in academic circles. Likewise, the practicing accountant has a right to expect the academic world to keep him informed of new developments which will have a significant impact on his profession.

Why Study Mathematics?

There are two main reasons why an accountant should study mathematics. First, the use of mathematics in problem solving requires a consistent logical approach. This approach may be thought of as including:

a) definition of a problem,
b) extraction of relevant information and representation of it in concise symbolic form,
c) use of logical deduction and/or selection of a known technique of mathematics to translate the initial representation into a suitable form from which a solution can be determined, and
d) extraction of the solution to the problem from this new form.

For example, one may set a problem up in the form of equations and then solve them. This process fosters clear thinking about problems and organized approaches to their solutions. To some degree, the subject matter of the problem is not as important as the process. Thus, it is perfectly reasonable for the future accountant to study high school geometry. This argument should not be extended too far, however. Even the professional mathematician can learn only a small fraction of the existing body of mathematical knowledge. Thus, the accountant must quickly come to the second basic reason for his study of mathematics.

This second reason is that certain areas of mathematics have useful applications in the study and in the practice of accounting. Accountants have found, for example, practical applications of certain topics in calculus (e.g., profit-volume analysis when the total revenue and/or total cost functions are nonlinear) and in linear algebra (e.g., simultaneous allocation of service department costs). These topics and others that are essential to an understanding of them are then legitimately included in the catalogue of material that the accountant might study. Of course, not every accountant need study everything in that catalogue.

We must be on guard against spurious reasons for studying mathematics, for these may lead to misallocations of the student's scarce time. One such reason is the well recognized "snob value" of mathematics. Within mathematics there is a pecking order, or scale of snobbery, with pure mathematicians as the high priests. Those, such as G. H. Hardy, reflect the extreme position by asserting that the pursuit of truth, elegance, and beauty in pure mathematics should not be sullied by any concern for applications.[2] Presumably, it is a lower form of animal, the applied mathematician, who will dirty his fingers in efforts to develop or adapt mathematics for practical uses. This pious concern for purity is attacked in a delightfully barbed passage from *Gulliver's Travels.*

> The learning of this people is very defective, consisting only in morality, history, poetry, and mathematics, wherein they must be allowed to excel. But the last of these is wholly applied to what may be useful in life, to the improvement of agriculture, and all mechanical arts; so that among us it would be little esteemed. And as to ideas, entities, abstractions, and transcendentals, I could never drive the least conception into their heads.[3]

We do not wish to go on record as being opposed to abstractions, but we do contend that the mere quality of abstractness is not, in itself, a virtue.

Another facet of the scale of snobbery in mathematics might be described as the "multiple integral syndrome." This suggests that those who can handle multiple integration are somehow superior to those who stopped at double integration and that these, in turn, outclass the single integrationists. Of course, there is no hope for anyone backward enough to have foregone calculus altogether. This attitude ignores the fact that the practical value of such knowledge declines sharply beyond the level of single integration. As teachers, we must avoid a tendency to put students through mathematical hoops simply because we may have learned to go through them ourselves. We would be far better advised to seek to develop or adapt mathematical techniques with specific relevance to accounting problems.

Corcoran argued that, without a familiarity with mathematics, one cannot understand the current literature of most business fields (p. 372). While there is undeniably some substance to this contention, it may prove to be a circular argument. If the literature fails to make clear how its mathematical content will provide some special insight or practical value, the reader should not feel apologetic about dismissing it after a brief inspection. We are currently in danger of equating the mathematical sophistication of an article or a book with its substantive quality. The number of mathematical symbols per page is a poor index of the worth of a piece of writing. In point of fact, too much or too complex mathematical content may substantially reduce the effective communication of ideas.

The primary purpose of symbology is to add clarity and precision to a discussion, and the symbols used should be as simple as needed to make the point. Of course, some areas require sophisticated mathematics; but these, too, should conform to this rule of maximum simplicity. We may expect editors to look for material whose mathematical content is consistent with their estimates of the backgrounds of their readership. If space does not permit an article to be self contained mathematically,

[2] *A Mathematician's Apology* (Cambridge University Press, 1967 (1940)).
[3] Jonathan Swift, *A Voyage to Brobdingnag*, Ch. 7.

or if a prior mathematical background is required, references should be made to supplementary sources. Corcoran, for example, fails to follow these guidelines in his derivation of the formula for the sum of the first n integers (p. 361). His approach is excessively complex and not self sufficient. He uses finite calculus and asks his readers to accept a theorem

$$\left(\Delta^{-1}ax^{(b)} = \frac{ax^{(b+1)}}{b+1}\Bigg]\right)$$

with no accompanying reference. He then asserts that

$$\sum_{x=1}^{n} x = \Delta^{-1}x^{(1)}\Bigg]_{1}^{n+1}$$

and, from the theorem, has little difficulty in producing the result. Such an approach can do real harm. It is easy to imagine a conscientious reader becoming discouraged at his failure to follow the argument, if he is unfamiliar with finite calculus. Not only might he abandon this article as being beyond his comprehension, but he might quickly reject subsequent articles with mathematical content. Our point is that this is all quite unnecessary, as it is much easier to prove the formula by use of a specific case of the procedure for summing n terms of an arithmetic progression, thus:

Let

$$S = 1 + 2 + 3 + \cdots + (n-1) + n$$

$$S = n + (n-1) + (n-2) + \cdots + 2 + 1$$

(Reversing the order)

Add; thus

$$2S = (n+1) + (n+1) + (n+1) + \cdots + (n+1) + (n+1)$$

(Now the right side contains n terms, each being $(n+1)$.)

Or

$$2S = n(n+1)$$

So

$$S = \frac{n(n+1)}{2}$$

In any event, comprehension of this proof is not essential to an understanding of the basic point Corcoran wishes to make.

Why Study Computers?

The basic reasons for studying computers are parallel to those for studying mathematics. First and foremost, of course, the computer is extremely useful in the solution of many problems confronting accountants and businessmen generally. However, study of computers, and particularly computer programming, also stimulates careful analysis of problems and approaches to their solutions; and it fosters the habit of organized problem solving. We believe the first of these reasons is clearly the more substantial. Practical computer applications of interest to the accountant are being developed with increasing frequency, and it will become progressively more damaging for him to remain uninformed about such new developments and opportunities. Corcoran's article recognizes the usefulness of computers for problem solving, but his discussion implicitly relegates the computer to the role of a computational device, a giant slide rule. It is in this respect that he has bound the computer's right hand behind its back. Computational convenience is not the feature that makes the computer so important to accountants and businessmen. Rather, it is the computer's ability to facilitate data processing in the broadest sense of the term. Computers provide the basis for modern attempts to make relevant information available to decision makers when, where, and how they want it; and these kinds of problems cannot be attacked by use of mathematical analysis.

An example is the now familiar airline reservation system. A combination of data processing units and communications capability allows insertion and retrieval of information from hundreds of inquiry stations fast enough to satisfy waiting customers. This is an efficient system for the airlines, as it allows them to provide better service and to maintain better control over their inventories of seats. Computer-based inventory control systems offer a similar illustration. The mathematics involved in the day-to-day operation of systems of this type is trivial and is not the essence of the system at all. (We should note in passing, however, that future inventory control systems will rely more on mathematically based techniques for forecasting and ordering, which will be implemented by means of the computer.)

Mathematicians may very reasonably regard the computer as a highly efficient calculator, if such is the way in which they have occasion to use it. This is not the way in which the computer is of principal value to accountants, however; and Corcoran errs in restricting his discussion to this role of the computer. A manager or an accountant properly views the computer as the central component of an information system, a system that comprises a variety of hardware, software, files, inputs, outputs, and people. It is inconceivable that an accountant today should be unfamiliar with such systems and with the computers that make them possible. We do not suggest that he needs a deep technical knowledge of computers, but he certainly should know their capabilities. Even such a mundane computer application as a payroll program offers opportunities for further managerial analyses. Management can quickly find answers to such questions as "What would happen to profits if wage rates were boosted 5 percent?" Such questions could also be answered through a manual payroll system, but the time and effort required would likely discourage the attempt. The computer system provides a readily accessible data base and programs for processing those data in a wide variety of ways. As broader based information systems become more common, this advantage of the computer over manual processing will be increased many times.

Corcoran alleges that an accountant cannot be an expert in information systems without knowing scientific (e.g., FORTRAN) and data processing (e.g., COBOL) programming languages and some minimal amount of mathematics (p. 360). While these abilities are commendable, to suggest that they are primary is to put the emphasis in the wrong place. It is much more important for the executive or the accountant to be imaginative about potential applications of computer systems and to learn to work well with systems analysts and programmers, who will convert his objectives into computer applications. Some skill in programming helps in conceiving applications and in communicating with computer specialists, but it is not essential for success in information systems.

Even in the area of solving mathematically formulated problems, Corcoran has not done the computer justice. The computer has three possible roles here.

(1) It may perform the calculations of a mathematical technique. Here the computer is doing exactly what the human problem solver would do: it is merely saving him the time and the tedium required to make the manual calculations. An example of this type of application is the simplex method of linear programming. A manual approach is sensible for the student who is just learning the method, for it will enable him better to understand the process involved. Once he understands it, however, there is no point to his repeating the manual process every time he wants to solve a problem. In this type of situation, the computer and mathematics are genuine partners. Mathematics pro-

vides the methodology of problem solving, while the computer does the actual computation. A computer user may call upon "canned" programs and subroutines for standardized mathematical and statistical calculations without even having to write the programs himself. One must be cautious here, however. The computer user must know enough mathematics to model his problem (as Corcoran observes on p. 365), to be certain that the assumptions underlying the mathematical technique apply to his example, and to interpret the output correctly. If he fails in these respects, he is guilty of computation without comprehension, a fault which may prove, at best, to be embarrassing.

(2) The computer may use a tedious method to solve a problem after it has been formulated in mathematical terms. This differs from the first role discussed above in that the computer departs from the manual mathematical solution process. The model is the same, but the computer substitutes a different method of processing. Corcoran's illustration of the newsboy problem, the minimization of opportunity cost (pp. 362–5), is an example of this role. Once the formula for the expected cost of stocking S papers has been established, it may be solved manually by calculus or it may be solved by the computer by successive evaluation of every alternative. Corcoran notes correctly that the computer solution technique, although less elegant, provides more information about the sensitivity of the optimal stocking policy. Corcoran's example of determining the probability distribution of demand during lead time (pp. 365–72) also fits into this class of problem, but there is an important difference. In the newsboy problem, both the manual and the computer solutions lead to the correct answer in a finite number of steps. In the lead-time demand problem, the mathematics of convolution theory yields the correct answer in a finite number of steps; but the computer solution does not. The computer implements a Monte Carlo analysis that reaches an approximation of the correct answer by an iterative process, and the accuracy of the approximation depends on the number of iterations. There is currently considerable interest in the validation of results of simulations.[4] How long should a simulation be run, and what confidence can one have in its results? The computer gives us a powerful capability for simulating systems, but we must be careful to appraise the results critically.

(3) The computer may use a problem-solving method that is not based upon mathematical formulation at all and that would be rejected as too tedious in the absence of a computer. Corcoran's illustration of summing digits (pp. 361–2) is an example of this. The significance of this illustration is that someone who has never heard of finite calculus, or even of the formula, can still get the desired result by having the computer perform the repetitive additions that any child could see would produce the correct answer.

We pointed out earlier that mathematics encourages a logical approach to problem solving and clear thinking generally. Similar claims can be made for computer programming. A strong argument for having accounting students learn programming is the orderly and analytical process that is essential to the completion of a successful program. In the process of writing a program to solve a problem, the student is likely to attain a much better understanding of the problem itself. Even though an accountant may never actually write a program in his professional work, a basic knowledge of programming makes him appreciate how important logical analysis is and enables him to have some

[4] Thomas H. Naylor and J. M. Finger, "Verification of Computer Simulation Models," *Management Science*, Vol. 14 (October 1967), pp. 92–101.

appreciation for the activities of the computer specialists with whom he will have to work from time to time. Further, it is not unusual for the programming of a particular application for the computer to reveal inconsistencies and inadequacies in past practices. Such disclosures provide a real benefit to management even before the computer application is running.

Good programming practices should be encouraged from the very start of a student's instruction. It is not enough for a program to accomplish its stated purpose, say, to produce a solution to a problem. It should do so in a manner that is general, efficient, creative, and clear.

Generality is that feature which ensures that a program will be able to solve as many problems of a particular type as possible. For example, if the parameters and the initial values of variables in a problem are read in by the program rather than being set within the program itself, that program will be capable of solving any similar problem. None of the programs in Corcoran's article could be run for different data values without replacing instructions and recompiling the program. In the case of the MONTCARL program (pp. 370–1), considerable revision would be required. This is undesirable and inefficient. A program for calculating depreciation may be used to illustrate the concept of generality. A single program could prepare depreciation schedules for a variety of assets, each with its own original cost, useful life, and salvage value, so long as an alphabetic description of each asset along with its relevant numeric values are read as input data. One computer run could then list each asset with its description and a depreciation schedule conforming to the method chosen for it. If desired, the same program could also test alternative depreciation methods and indicate which would be most advantageous for tax purposes.

A program is efficient to the extent that it accomplishes its intended purpose with a minimum of time and effort on the part of the computer user and without excessive use of costly computer time and storage space. There are programming considerations that can save computer time without complicating a person's use of the program in any way. For example, unnecessary statements should be kept out of program loops, especially the innermost of a set of nested loops, to avoid recomputation of a value that has not changed. Pretested routines should be used to the extent possible, for they are designed to be efficient and are already error free. In the earlier example of summing the first n integers, the simple method of adding the terms would be less desirable than using the formula if n were large or if the process were to be repeated many times. These desirable characteristics of efficient programming are independent of the machine or the compiler used.

Creativity in programming is probably the single most valuable feature for purposes of general educational development, and it is just as hard to define as in other areas of creative endeavor. In varying degrees, it is the capacity to generalize, to solve problems in "neat" ways, and to recognize that certain unpromising problems are amenable to computer solution. Examples of this kind of activity can be found in a number of computer solutions which appear in the "Mathematical Games" section of *Scientific American*. One such problem (not from *Scientific American*) is the commercially distributed puzzle known as "instant insanity." It consists of four cubes, the sides of which are variously colored blue, green, red, and white. The object is to stack the four cubes so that each of the four colors appears once only on each side of the stack. A program to do this is illustrated in Exhibit I. The output of this program shows that, for the par-

Exhibit I

```
C
C  THIS PROGRAM RUINS AN INNOCENT HALF HOUR SPENT WRESTLING
C  WITH THE PUZZLE CALLED INSTANT INSANITY. THE COLOUR CODE
C  IS RED=1, GREEN=2, WHITE=3, BLUE=4. WE HAVE CALLED CUBE ONE
C  THE CUBE WITH TWO RED AND TWO WHITE FACES, CUBE TWO HAS
C  THREE REDS, CUBE THREE HAS TWO GREENS AND TWO WHITES, AND
C  CUBE FOUR HAS TWO GREENS AND TWO BLUES. THE DATA RECORDS
C  THE COLOURS OF EACH CUBE IN ORDER, FRONT, RIGHT SIDE, BACK,
C  LEFT SIDE, TOP AND BASE.
C  THIS PREAMBLE IS MUCH TOO LONG. IT IS A PITY THE STUPID
C  COMPILER CANT LEARN TO CUT VERBOSITY IN MID SENT * URGHR * *.
C
      DIMENSION N(192), IT(4), MA(24)
C
C  READ 6 COLOURS FOR EACH CUBE
C
      READ (5, 1) (MA(I), I=1, 24)
    1 FORMAT (24I1)
C
C  HEADING AND INPUT DATA
C
      WRITE (6, 20) (I, I=1, 4), (MA(I), I=1, 24), (I, I=1, 4)
   20 FORMAT ('1' T20, 'INSTANT INSANITY PUZZLE'/ / / /T30, 'RED=1'
     1/ /T30, 'GREEN=2'/ /T30, 'WHITE=3'/ /T30, 'BLUE=4'/ / / /T10,
     2'F=FRONT'/ /T10, 'R=RIGHT'/ /T10, 'B=BACK'/ /T10, 'L=LEFT'/ /
     3T10, 'T=TOP'/ /T10, 'BS=BASE'/ /T5, 4 ('CUBE' I2, 8X)/ /T2,
     44 ('F R B L T BS' 2X)/ /T2, 4 (6(I1, 1X), 2X), 5 (/), T20,
     5'SOLUTIONS FOLLOW'/ /T10, 4('CUBE' I2, 5X)/ /T9, 4('F R B L'
     64X)/)
C
C  FOR EACH CUBE, GENERATE ALL POSSIBLE VALUES FOR THE SIDES
C  RECORDED AS FRONT, RIGHT, BACK AND LEFT. THE THREE AXES OF
C  ROTATION LEAD TO TWELVE SETS PER CUBE EACH WITH FOUR
C  DATA VALUES
C
      L=1
      DO 10 KA=1, 19, 6
      JT=0
      KT=1
      KB=KA+3
      DO 11 KS=KA, KB
      IT (KT)=MA(KS)
      KT=KT+1
   11 CONTINUE
      GO TO 12
   13 IT(2)=MA(KB+1)
      IT(4)=MA(KB+2)
      GO TO 12
   14 IT(1)=MA(KB)
      IT(3)=MA(KA+1)
   12 LL=L+15
      K=1
      J=0
      DO 15 I=L, LL
      N(I)=IT(K)
      J=J+1
      IF(J.EQ.4)GO TO 16
      K=K+1
      IF(K.EQ.5)K=1
      GO TO 15
   16 J=0
      IF(K.EQ.1)K=3
      K=6-K
   15 CONTINUE
      L=L+16
      JT=JT+1
      IF(JT-2)13, 14, 10
   10 CONTINUE
      KQ=0
```

```
C
C   TEST FOUR NUMBERS FROM A CUBE AGAINST CORRESPONDING NOS
C   FROM OTHER CUBES. REJECT IF ANY TWO MATCH.
C
C
C   CUBE 1 AND CUBE 2
C
       DO 2 I=1, 45, 4
       DO 3 J=49, 93, 4
       DO 4 M=1, 4
       IF (N(I+M-1).EQ.N(J+M-1))GO TO 3
     4 CONTINUE
C
C   CUBE 1 AND 2 WITH CUBE 3
C
       DO 5 K=97, 141, 4
       DO 6 M=1, 4
       IF(N(I+M-1).EQ.N(K+M-1))GO TO 5
       IF(N(J+M-1).EQ.N(K+M-1))GO TO 5
     6 CONTINUE
C
C   CUBE 1 AND 2 AND 3 WITH CUBE 4
C
       DO 7 L=145, 189, 4
       DO 8 M=1, 4
       IF(N(I+M-1).EQ.N(L+M-1))GO TO 7
       IF(N(J+M-1).EQ.N(L+M-1))GO TO 7
       IF(N(K+M-1).EQ.N(L+M-1))GO TO 7
     8 CONTINUE
       WRITE (6, 9)(N(I+M-1), M=1, 4), (N(J+M-1), M=1, 4), (N(K+M-1),
      1M=1, 4), (N(L+M-1), M=1, 4)
     9 FORMAT (T9, 4 (4(I1, 1X), 3X)/ /)
       KQ=KQ+1
     7 CONTINUE
     5 CONTINUE
     3 CONTINUE
     2 CONTINUE
       IF (KQ.EQ.0) WRITE (6, 21)
    21 FORMAT (T20, 'THERE ARE NO SOLUTIONS')
       END
```

```
                        INSTANT INSANITY PUZZLE
                                 RED=1
                                 GREEN=2
                                 WHITE=3
                                 BLUE=4
      F=FRONT
      R=RIGHT
      B=BACK
      L=LEFT
      T=TOP
      BS=BASE
  CUBE 1          CUBE 2          CUBE 3          CUBE 4
F R B L T BS    F R B L T BS    F R B L T BS    F R B L T BS
1 2 3 3 1 4     1 4 1 3 1 2     1 3 2 4 2 3     1 2 4 2 4 3

             SOLUTIONS FOLLOW
      CUBE 1   CUBE 2   CUBE 3   CUBE 4
      F R B L  F R B L  F R B L  F R B L
      3 1 2 4  4 2 3 1  2 3 1 2  1 4 4 3
      1 2 4 3  2 3 1 4  3 1 2 2  4 4 3 1
      2 4 3 1  3 1 4 2  1 2 2 3  4 3 1 4
      4 3 1 2  1 4 2 3  2 2 3 1  3 1 4 4
```

ticular set of blocks sold commercially, the solution is unique (if one discounts simple rotation of the entire stack, as this is adjudged to be a single solution). The program systematically tests all possible combinations and prints out solutions. This approach is available to a human, but he is not likely to concentrate long enough to avoid erroneous repetitions; and he will soon abandon it as unsatisfactory. An intelligent player can make logical inferences which reduce the solution space to be searched and can quite quickly find a solution, but he does not know if it is unique and he must start afresh with a new combination of colors. The computer program will solve any color combination as fast as the data can be entered and run. (We confess that this application does not stand up very well to our argument that students should concentrate on matters that have practical utility. It might just about justify itself as an example of misapplied ingenuity, however.)

Finally, the clarity of a program is a measure of the ease with which someone other than the programmer can understand exactly what the program does and how it may be used. Clarity is enhanced by adhering to standard procedures and by good program documentation. Documentation may take the forms of comments within the body of the program itself, flow charts, and supporting descriptive and narrative materials. Attention to all four of the criteria for good programming will not only make the student a better programmer but will also make him more careful and more imaginative in problem solving generally.

Corcoran's article mentions a talented student who became fascinated by the computer and turned his attention more to it than to mathematics and, thus, was captured by "the enemy" (pp. 359–60). We should reflect on why some students get "the computer bug" almost to the point of an obsession. It certainly is not a desire to escape the rigor of mathematics. We would conjecture that the enthusiasm comes from a unique problem-solving environment. Successive attempts are quickly and thoroughly evaluated by an impartial and anonymous entity. New opportunities mushroom in applications, languages, and hardware—all waiting to be explored. For the enthusiast, it might be a major problem to retain the proper perspective and not endanger his academic standing by neglecting all else. Such dangers should be avoided, but they are impressive testimony to the fascination that the computer field holds for bright students.

How Much of Each Should One Study?

This is the critical practical question that accounting educators must seek to answer. There is no need for the answer to be the same for every student. To the extent that he has a choice, each student's abilities, interests, previous studies, experiences in various courses, and assessment of potential values will lead him to his own decision as to how far he goes into each of these important areas of study. Unquestionably, he must do more than dabble at the surface of each; but we cannot generalize as to an optimal mix of computers and mathematics.

Corcoran argues that an ability to work with mathematics is essential to anyone seeking new approaches to managerial problems (p. 372). We contend that exactly the same could be said of the computer. Indeed, the computer offers just as much promise as mathematics in the search for creative innovations in management. So, probably, do the behavioral sciences. Another reason for favoring mathematics over the computer in Corcoran's judgment is that mathematics are more demanding and intellectually rewarding (p. 372). This is a matter of opinion and, perhaps, of taste. We are not convinced, however, that the study of a highly structured and rigorous discipline is any more demanding than an investigation of an emerging discipline with many uncharted areas. And rewards are to be found in any subject in which the student takes a genuine interest. Intellectual rigor has merit, but there is no reason to assert that it is a higher criterion in the design of an educational program than is practical utility. The relevance of the computer to the future of an accountant or a manager is at least as great as that of mathematics. Accounting instructors should encourage their students to become actively involved with both subjects. Further, they should assume a professional responsibility to ensure that the relevant applications of both are incorporated throughout the students' subsequent accounting courses.

Mathematics and computers make their own distinctive demands on creativity and ingenuity, and the talented person will use each as the situation warrants. Often, a combination of the two will be best. Students and practitioners would be unwise to follow too strong a bias in either direction.